**COEN 161 Web Programming 1 Fall 2015**

**Lab 4\_Part 1 (30 pts)**

**Objectives**

* Learn to debug JavaScript programs using Chrome Developer Tools.

Create a directory called Lab4 under your Labs directory (under coen161).

Copy all the files in the zipped file, Lab4, into this directory.

In this part, you are shown how to debug simple JavaScript programs by using breakpoints.

**How to access Chrome Developer Tools:**

* Open Chrome web browser. To access the Developer Tools select the **Wrench menu** http://chromedevtools.googlecode.com/svn-history/r421/trunk/tutorials/breapoints/images/wrench.png at the top-right of your browser window, then select **Tools**-> **Developer tools**, or use the **Control-Shift-I** shortcut.

## Overview: Debugging With Breakpoints

A breakpoint is an **intentional** stopping or pausing place in a script, put in place for debugging purposes. We will look at **debugging** JavaScript using breakpoints set on JavaScript code.

### Workflow

* First, open the Developer Tools by hitting the **Control-Shift-I** shortcut. Click the **line gutter** to set a breakpoint for that line of code.
* All the breakpoints you have set appear under **Breakpoints** in the right-hand sidebar. Breakpoints can be enabled or disabled using the checkboxes in this sidebar.
* Clicking on the breakpoint entry jumps to the **highlighted line** in the source file. You can **set** one or more breakpoints in one or more scripts.
* Right click on the line number where the breakpoint is set to reveal the breakpoint menu (indicator). The breakpoint indicator menu has several options including "**Continue to here**", "**Remove Breakpoint**", "**Edit Breakpoint**", "**Disable Breakpoint**".

### Steps to set breakpoint:

* Open the Developer Tools by hitting the **Control-Shift-I** shortcut
* Select the Sources tab.
* Select Show Navigator icon.
* Click the little folder icon on the far left.
* Navigate to the source file where you want to set the break point.
* Find the line where you want the breakpoint, then click the line number (in line gutter) to set the breakpoint. A little highlighted triangle will appear (you can use the **Control-G** shortcut to go to a line in a large file).
* Refresh the html page in your browser by the either clicking the refresh icon in Browser toolbar or by pressing **F5**
* You should stop on the breakpoint
* Hover over the source code to inspect local and global variables
* Click on **Step into next function call** (F11) to execute the next line of code.
* Examine the value of variables by hovering over the source code.
* You can also use **Step over next function call** (F10), **Step out of current function** (<Shift>+F11) and **Deactivate Breakpoints** from the menu on the right side.
* Clicking **Continue** http://chromedevtools.googlecode.com/svn-history/r421/trunk/tutorials/breapoints/images/debuggerContinue.png button or hit **F8** in Developer Tools window will resume execution of the program.

All the breakpoints you have set appear under **Breakpoints** in the right-hand sidebar. Clicking on the entry jumps to the highlighted line in the source file.

**Exercise:** **(30 points)**

Follow the steps as given above and debug the JavaScript files given in **Lab4\_Part1** folder. The programs have logical errors. Debug the code so that you get the output as mentioned below.

1. **ConditionalStmt.html:**

**Should display message based on the following conditions,**

* Display “What are you doing up so late?” if current time is less than 5 am
* Display “Good Morning!” if current time between 5 am and 9 am
* Display "No surfing during working hours!" if current time between 9 am and 5 pm
* Otherwise Display "Good Evening!"

1. **factorial.html**

The program should display a table with factorial of integers from 1 to 10. The program uses recursion.

1. **findMax.html**

Display the max of three numbers which are obtained as input from the user (window prompt).

Visit <https://developers.google.com/chrome-developer-tools/docs/javascript-debugging> for more information on JavaScript Debugging in Chrome using Developer Tools.

**Firebug (Firefox)**

Firebug is a debug tool for Firefox browser. JavaScript code can be debugged using this tool. The extension can be downloaded and installed at <http://getfirebug.com/>

Information about Firebug extension for Firefox can be found at <http://getfirebug.com/wiki/index.php/Script_Panel>

*Reference:* [*https://developers.google.com/chrome-developer-tools/*](https://developers.google.com/chrome-developer-tools/)